FIG. 1

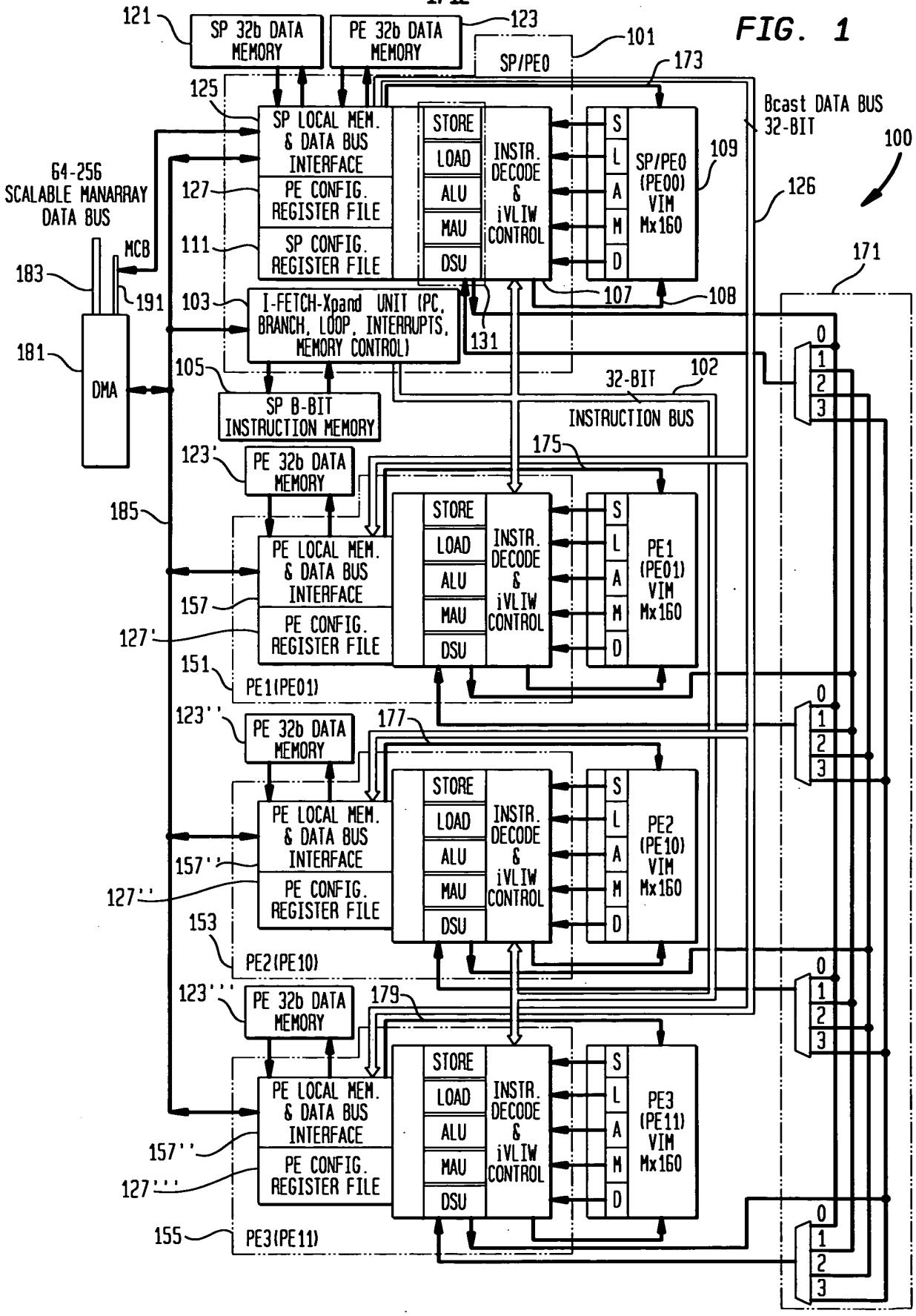

121
SP 32b DATA MEMORY
123
PE 32b DATA MEMORY
101
SP/PE0
173
Bcast DATA BUS 32-BIT
100
125
SP LOCAL MEM. & DATA BUS INTERFACE
STORE
LOAD
ALU
MAU
DSU
INSTR. DECODE & iVLIW CONTROL
S
L
A
M
D
SP/PE0 (PE00) VIM Mx160
109
126
64-256 SCALABLE MANARRAY DATA BUS
127
PE CONFIG. REGISTER FILE
111
SP CONFIG. REGISTER FILE
MCB
183
191
103
I-FETCH-Xpand UNIT (PC, BRANCH, LOOP, INTERRUPTS, MEMORY CONTROL)
131
107
108
171
181
DMA
105
SP B-BIT INSTRUCTION MEMORY
32-BIT
102
INSTRUCTION BUS
175
123'
PE 32b DATA MEMORY
185
PE LOCAL MEM. & DATA BUS INTERFACE
157
127'
PE CONFIG. REGISTER FILE
PE1 (PE01) VIM Mx160
151
PE1(PE01)
123''
PE 32b DATA MEMORY
177
157''
127''
PE2 (PE10) VIM Mx160
153
PE2(PE10)
123'''
PE 32b DATA MEMORY
179
157'''
127'''
PE3 (PE11) VIM Mx160
155
PE3(PE11)
0
1
2
3

FIG. 2

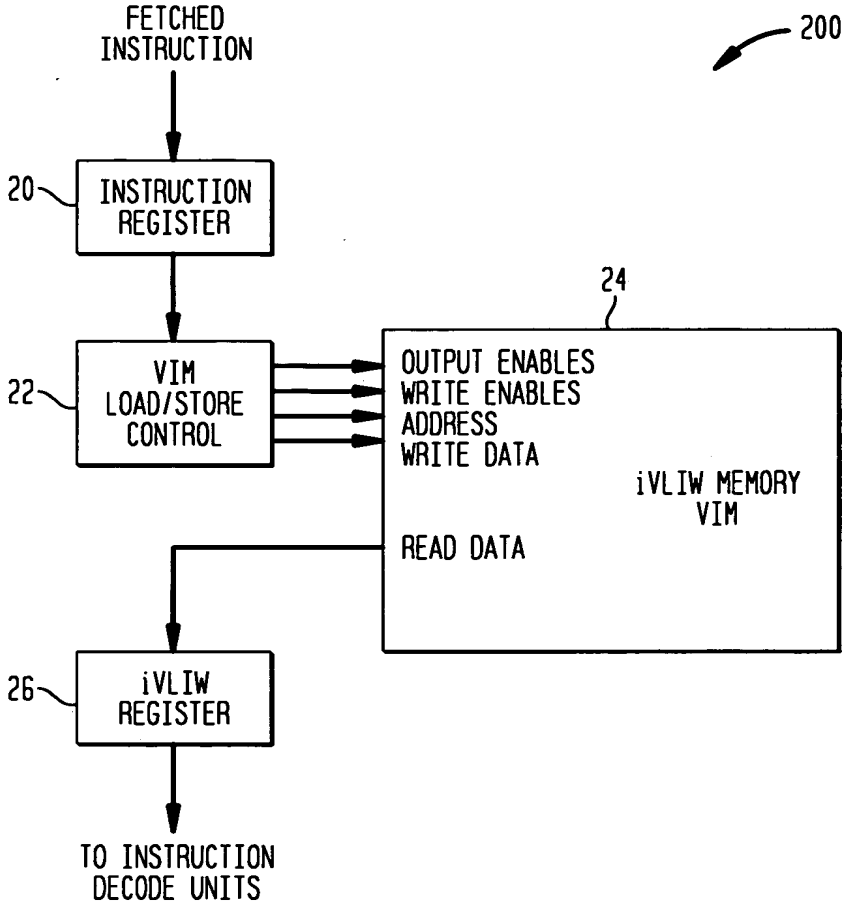

FETCHED INSTRUCTION
200
20
INSTRUCTION REGISTER
24
22
VIM LOAD/STORE CONTROL
OUTPUT ENABLES
WRITE ENABLES
ADDRESS
WRITE DATA
iVLIW MEMORY VIM
READ DATA
26
iVLIW REGISTER
TO INSTRUCTION DECODE UNITS

FIG. 3

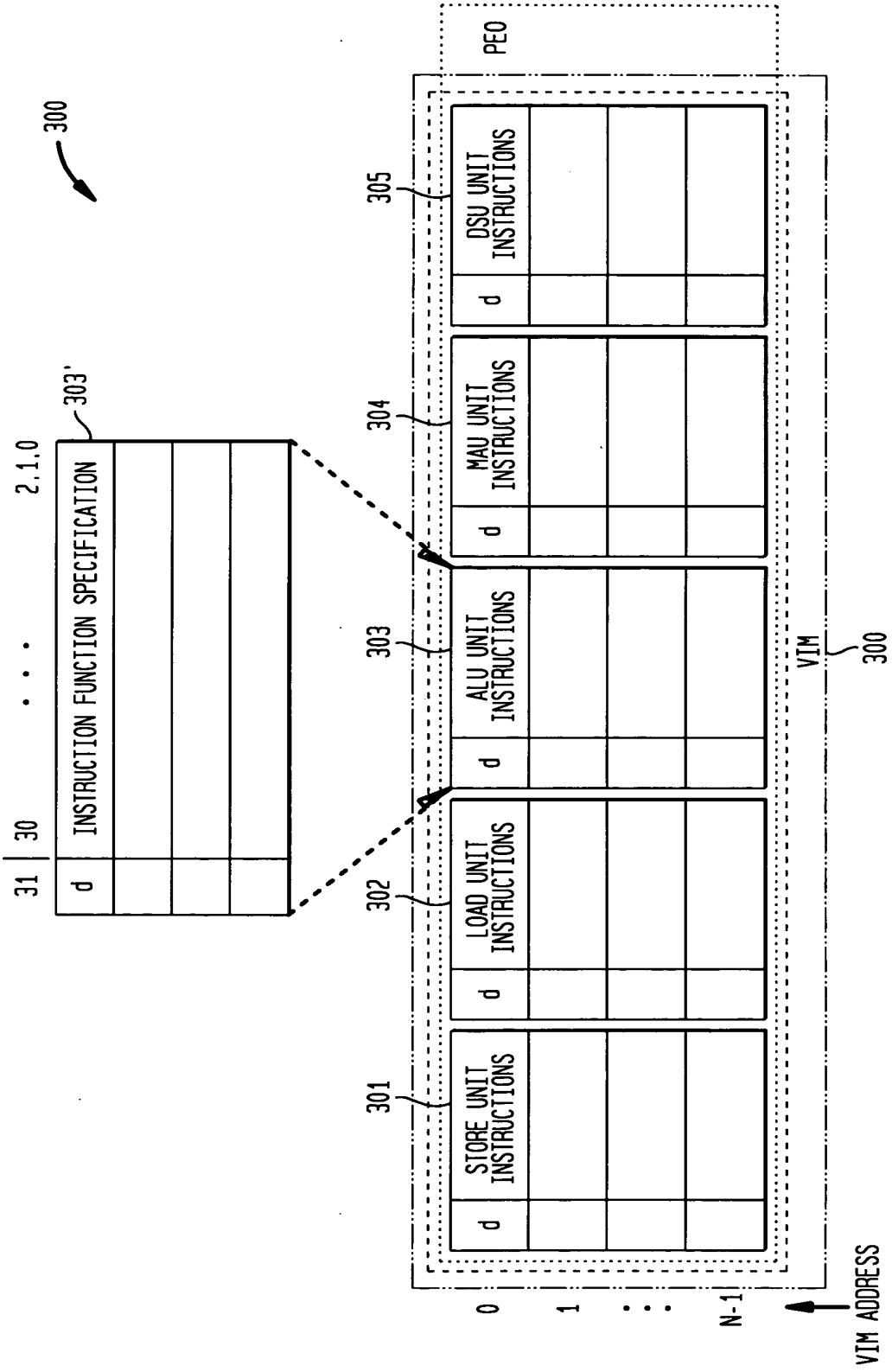

300
PEO
305
DSU UNIT INSTRUCTIONS
304
MAU UNIT INSTRUCTIONS
303
ALU UNIT INSTRUCTIONS
302
LOAD UNIT INSTRUCTIONS
301
STORE UNIT INSTRUCTIONS
d
VIM
300
0
1
N-1
VIM ADDRESS
303'
INSTRUCTION FUNCTION SPECIFICATION
31
30
2.1.0

FIG. 4A

LV1- LOAD/MODIFY VLIW ~400

Encoding ~410

411

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Group		S/P	CtrlOp				0	UAF		InstrCnt				0	0	0	SU	LU	ALU	MAU	DSU	Vb	0	VIMOFFS							

Syntax/Operation ~420

Instruction	Operands	Operation
LV1.[SP]	V[01], VIMOFFS, InstrCnt, d=(SLAMD), F=[AMD]	(V[01]+VIMOFFS)[SU] ←disable if (d=S) (V[01]+VIMOFFS)[LU] ←disable if (d=L) (V[01]+VIMOFFS)[ALU] ←disable if (d=A) (V[01]+VIMOFFS)[MAU] ←disable if (d=M) (V[01]+VIMOFFS)[DSU] ←disable if (d=D) (V[01]+VIMOFFS)[UAF] ←ALU if (F=A or F=) (V[01]+VIMOFFS)[UAF] ←MAU if (F=M) (V[01]+VIMOFFS)[UAF] ←DSU if (F=D) if (instrCnt > 0 Load next InstrCnt instructions into (V[01]+VIMOFFS)

FIG. 4B

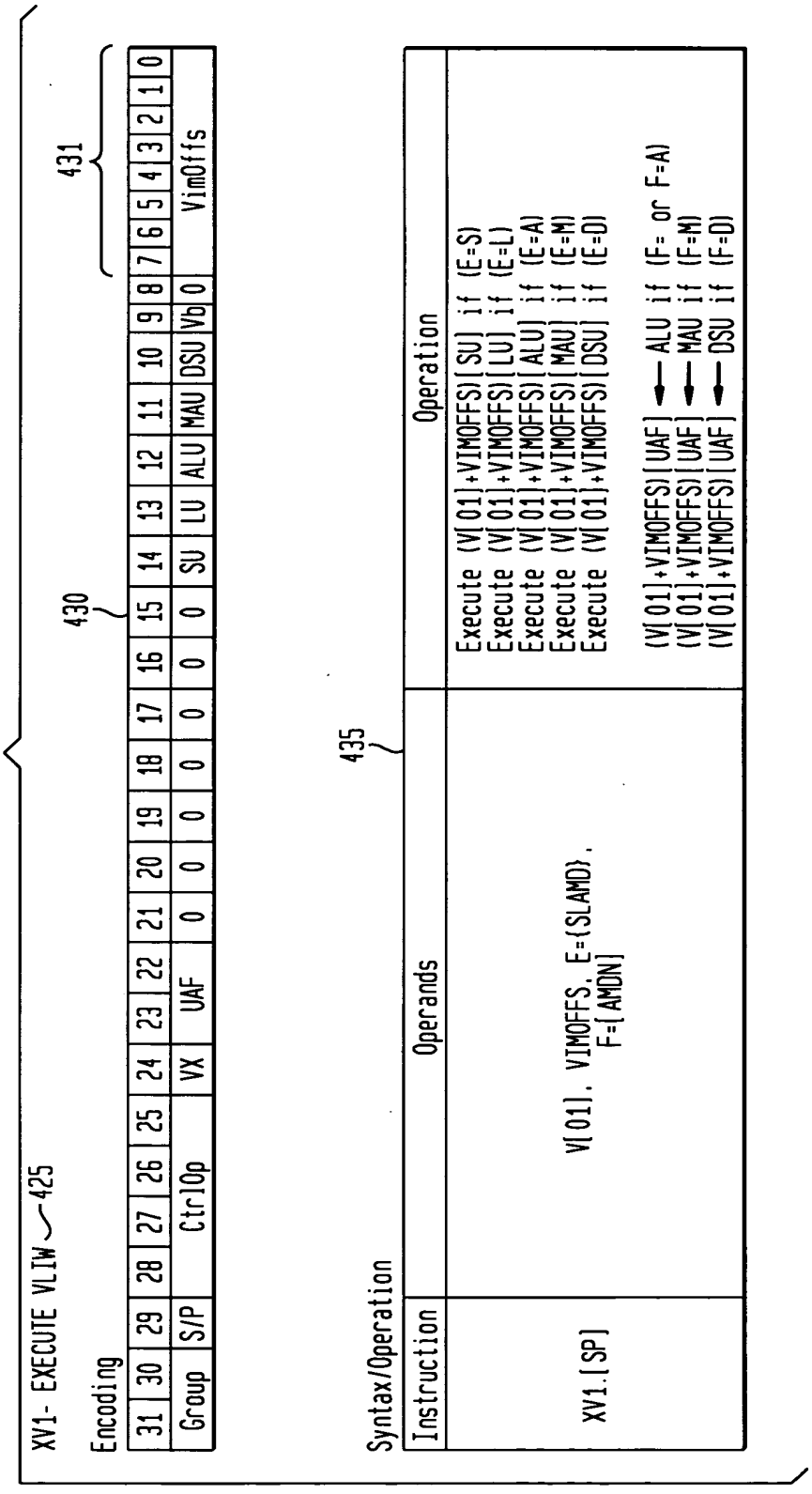

XV1- EXECUTE VLIW
425
Encoding
430
431
31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
Group S/P Ctrl0p VX UAF 0 0 0 0 0 0 0 SU LU ALU MAU DSU Vb 0 VimOffs
Syntax/Operation
435
Instruction
Operands
Operation
XV1.[SP]
V[01], VIMOFFS, E={SLAMD}, F={AMDN]
Execute (V[01]+VIMOFFS)[SU] if (E=S)
Execute (V[01]+VIMOFFS)[LU] if (E=L)
Execute (V[01]+VIMOFFS)[ALU] if (E=A)
Execute (V[01]+VIMOFFS)[MAU] if (E=M)
Execute (V[01]+VIMOFFS)[DSU] if (E=D)
(V[01]+VIMOFFS)[UAF] → ALU if (F= or F=A)
(V[01]+VIMOFFS)[UAF] → MAU if (F=M)
(V[01]+VIMOFFS)[UAF] → DSU if (F=D)

FIG. 4C

LV2- LOAD/MODIFY VLIW- 2 ~ 455

Encoding

450

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Group		S/P	CtrlOp				0	LI	d	InstrCnt				UnitVIM			0	0	0	0	0	Vb	0	VimOffs							

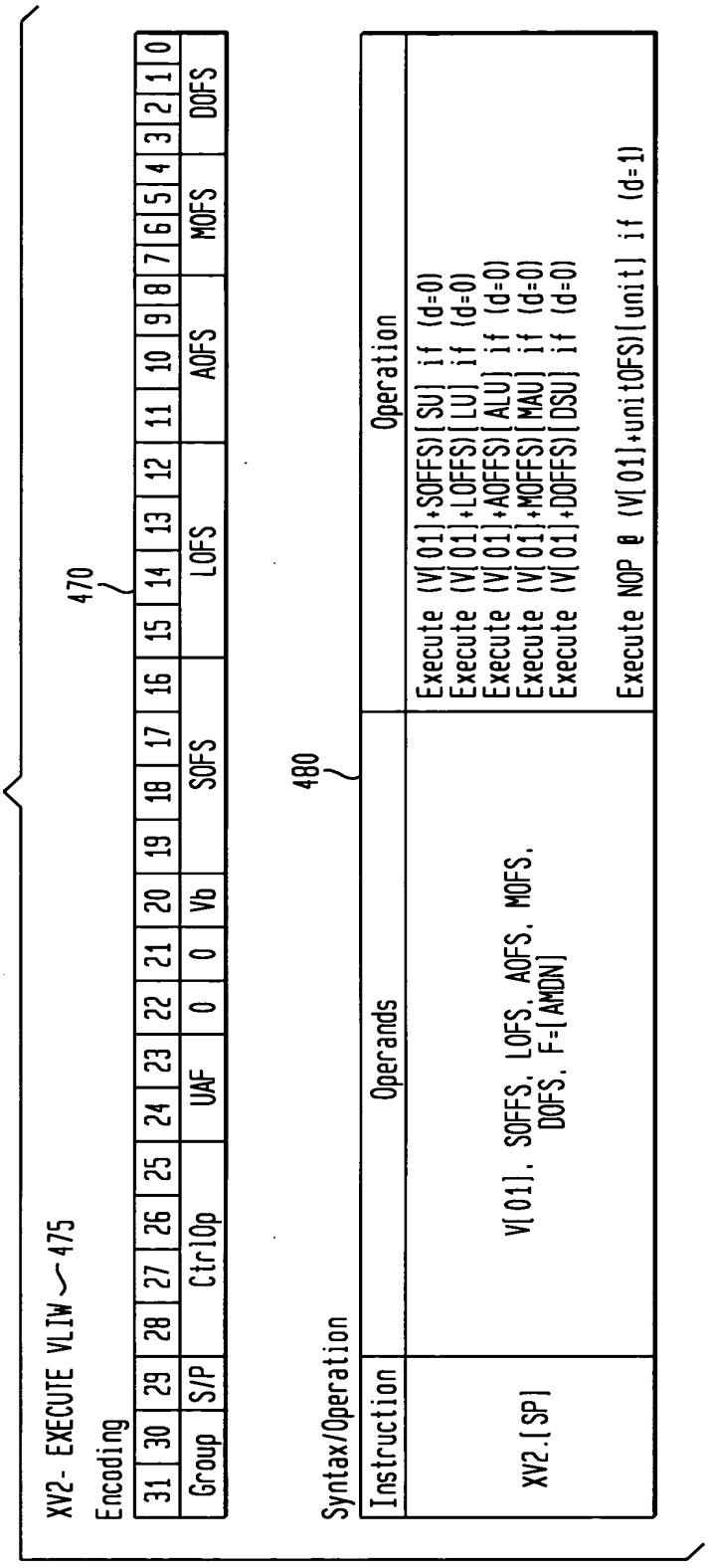
Syntax/Operation

460

Instruction	Operands	Operation

LV2.[SP]	LI, u=UnitVIM, V[01], VIMOFFS, InstrCnt, d	if (LI=0) Load disable bit only
		disable bit @ (V[01]+VIMOFFS)[UnitVIM] ← d
		
		if (LI=1) Load instructions
		disable bit @ (V[01]+VIMOFFS)[UnitVIM] ← d
		
		Load next InstrCnt instructions into
		(V[01]+VIMOFFS)[UnitVIM] ← 1st Instruction following LV2
		
		(V[01]+VIMOFFS)+1)[UnitVIM] ← 2nd Instruction following LV2
		⋮
		(V[01]+VIMOFFS)+InstrCnt)[UnitVIM] ← (InstrCnt)th Instruction following LV2
		
		InstrCnt is a binary coded number, 0 thru F, that represents from 1 to 16 instructions that can be loaded into up to 16 consecutive UnitVIM locations

FIG. 4D


XV2- EXECUTE VLIW
475
Encoding
470
31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
Group S/P Ctrl Op UAF 0 0 Vb SOFS LOFS AOFS MOFS DOFS
480
Syntax/Operation
Instruction
Operands
Operation
XV2.[SP]
V[01], SOFFS, LOFS, AOFS, MOFS, DOFS, F=[AMDN]
Execute (V[01]+SOFFS)[SU] if (d=0)
Execute (V[01]+LOFFS)[LU] if (d=0)
Execute (V[01]+AOFFS)[ALU] if (d=0)
Execute (V[01]+MOFFS)[MAU] if (d=0)
Execute (V[01]+DOFFS)[DSU] if (d=0)
Execute NOP @ (V[01]+unitOFS)[unit] if (d=1)

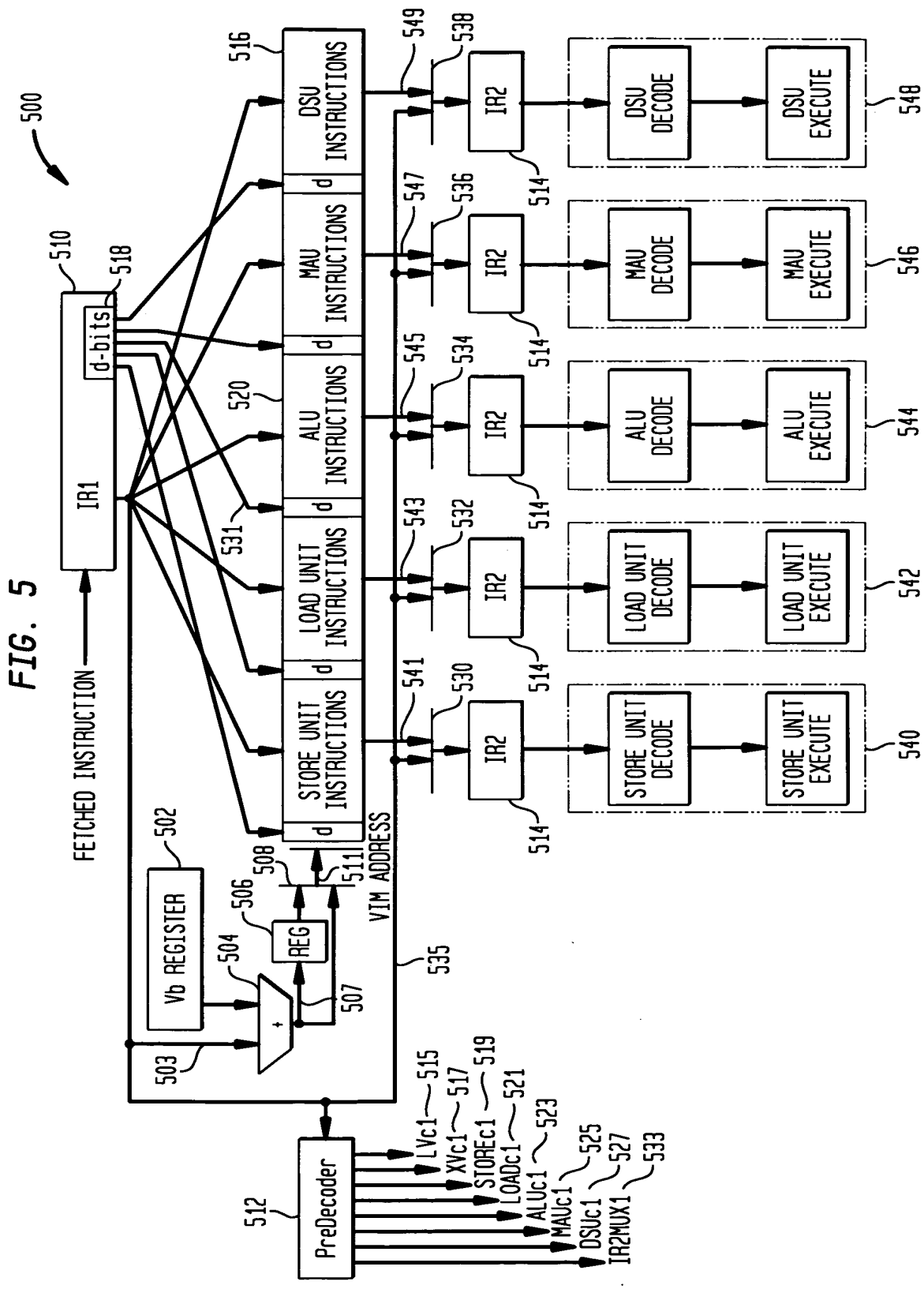
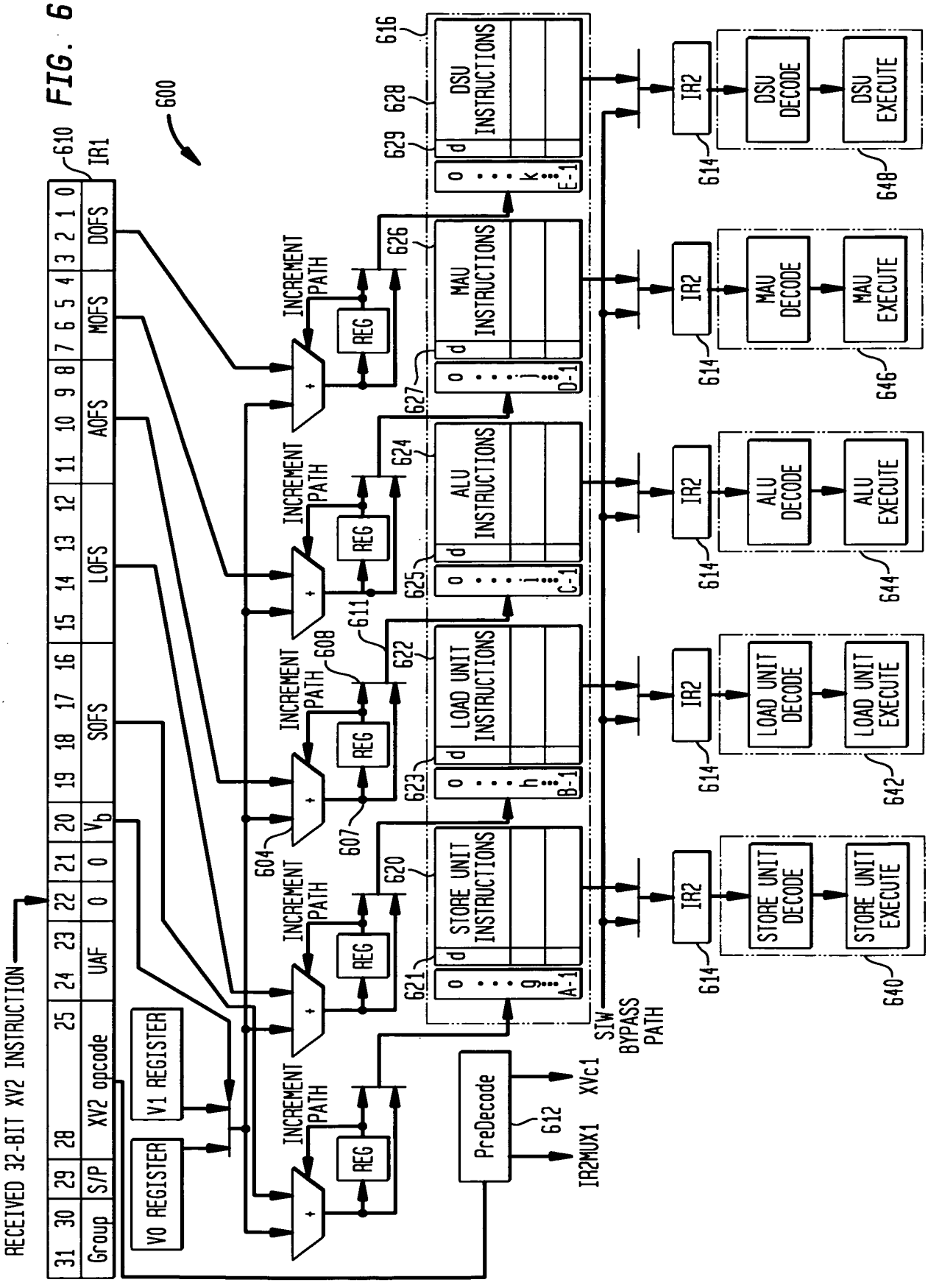

500
FETCHED INSTRUCTION
IR1
510
d-bits
518
516
DSU INSTRUCTIONS
MAU INSTRUCTIONS
520
ALU INSTRUCTIONS
531
LOAD UNIT INSTRUCTIONS
STORE UNIT INSTRUCTIONS
d
502
Vb REGISTER
504
503
508
506
REG
511
507
VIM ADDRESS
535
549
538
547
536
545
534
543
532
541
530
IR2
514
DSU DECODE
DSU EXECUTE
548
MAU DECODE
MAU EXECUTE
546
ALU DECODE
ALU EXECUTE
544
LOAD UNIT DECODE
LOAD UNIT EXECUTE
542
STORE UNIT DECODE
STORE UNIT EXECUTE
540
512
PreDecoder
LVc1
XVc1
STOREc1
LOADc1
ALUc1
MAUc1
DSUc1
IR2MUX1
515
517
519
521
523
525
527
533

FIG. 5

FIG. 6


600
RECEIVED 32-BIT XV2 INSTRUCTION
31 30 29 28 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
Group S/P XV2 opcode UAF 0 0 Vb SOFS LOFS AOFS MOFS DOFS
610 IR1
V0 REGISTER
V1 REGISTER
INCREMENT PATH
REG
604
607
608
611
PreDecode
612
IR2MUX1
XVc1
STW BYPASS PATH
STORE UNIT INSTRUCTIONS
LOAD UNIT INSTRUCTIONS
ALU INSTRUCTIONS
MAU INSTRUCTIONS
DSU INSTRUCTIONS
620
621
622
623
624
625
626
627
628
629
IR2
614
STORE UNIT DECODE
STORE UNIT EXECUTE
640
LOAD UNIT DECODE
LOAD UNIT EXECUTE
642
ALU DECODE
ALU EXECUTE
644
MAU DECODE
MAU EXECUTE
646
DSU DECODE
DSU EXECUTE
648

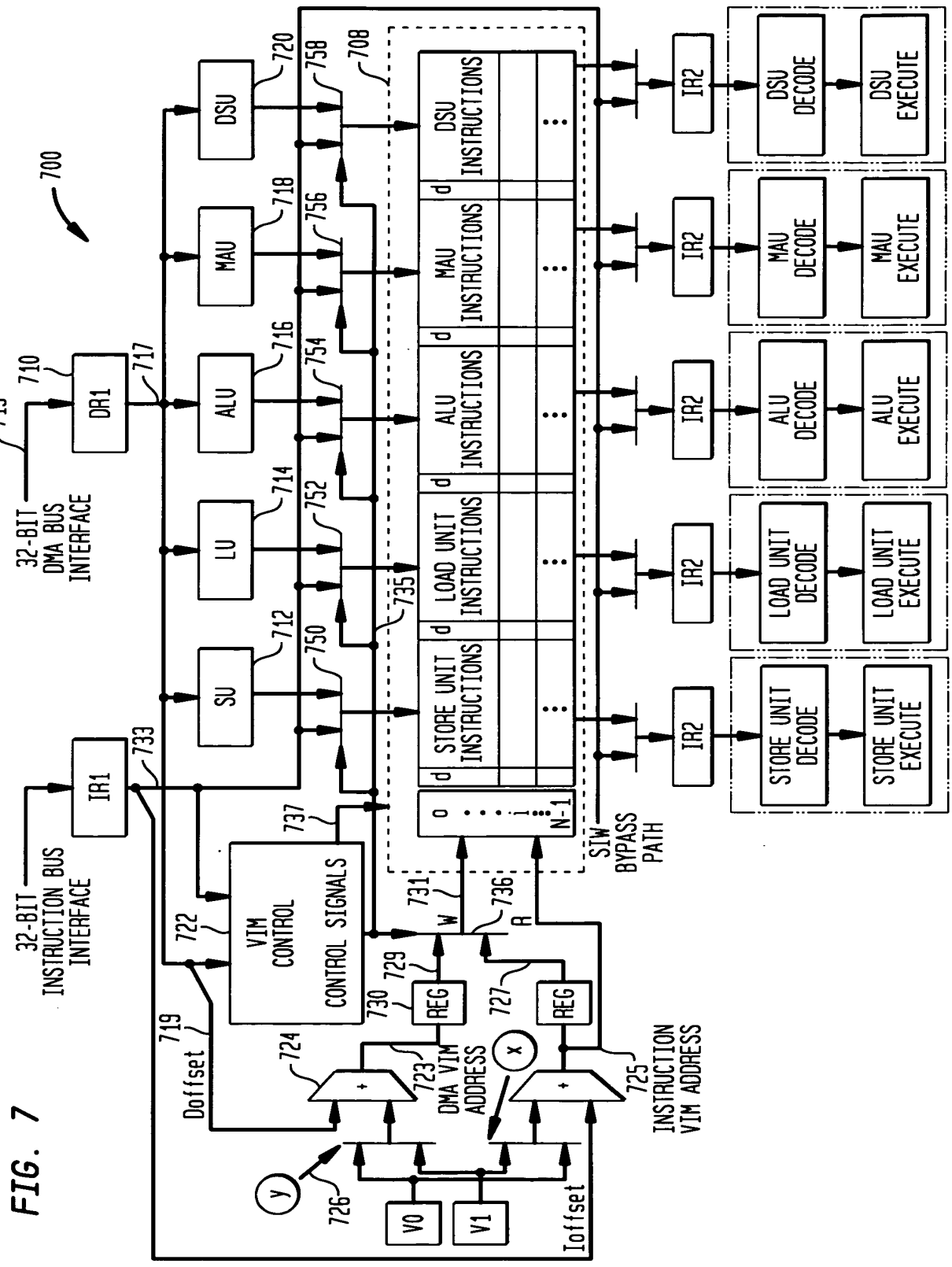

700
32-BIT INSTRUCTION BUS INTERFACE
32-BIT DMA BUS INTERFACE
IR1
DR1
SU
LU
ALU
MAU
DSU
STORE UNIT INSTRUCTIONS
LOAD UNIT INSTRUCTIONS
ALU INSTRUCTIONS
MAU INSTRUCTIONS
DSU INSTRUCTIONS
IR2
STORE UNIT DECODE
STORE UNIT EXECUTE
LOAD UNIT DECODE
LOAD UNIT EXECUTE
ALU DECODE
ALU EXECUTE
MAU DECODE
MAU EXECUTE
DSU DECODE
DSU EXECUTE
SIW BYPASS PATH
VIM CONTROL
CONTROL SIGNALS
REG
REG
DMA VIM ADDRESS
INSTRUCTION VIM ADDRESS
Doffset
Ioffset
V0
V1
W
R

FIG. 7

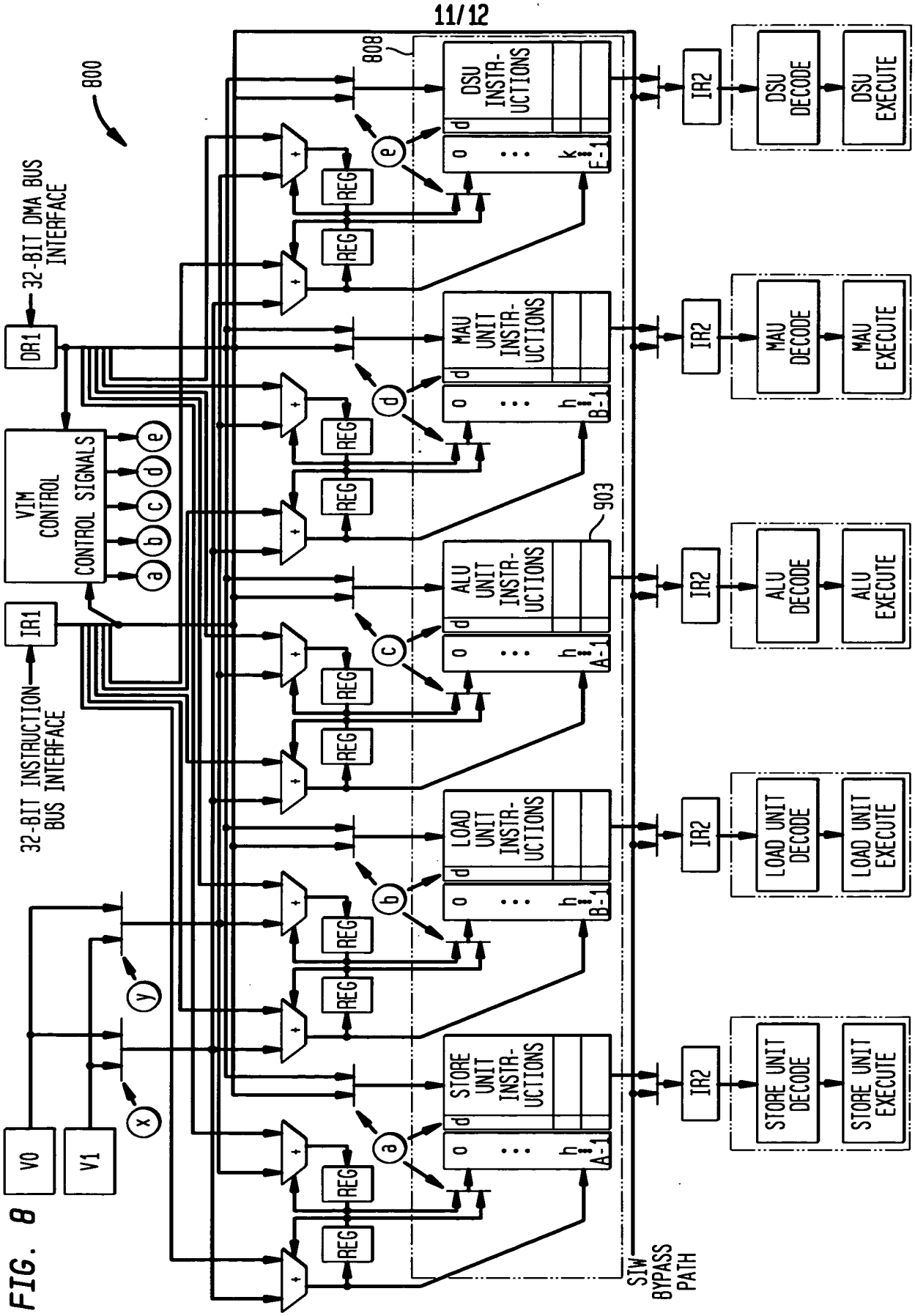

800
32-BIT DMA BUS INTERFACE
DR1
VIM CONTROL
CONTROL SIGNALS
IR1
32-BIT INSTRUCTION BUS INTERFACE
V0
V1
808
903
DSU INSTR-UCTIONS
MAU UNIT INSTR-UCTIONS
ALU UNIT INSTR-UCTIONS
LOAD UNIT INSTR-UCTIONS
STORE UNIT INSTR-UCTIONS
REG
IR2
DSU DECODE
DSU EXECUTE
MAU DECODE
MAU EXECUTE
ALU DECODE
ALU EXECUTE
LOAD UNIT DECODE
LOAD UNIT EXECUTE
STORE UNIT DECODE
STORE UNIT EXECUTE
SIW BYPASS PATH

FIG. 8

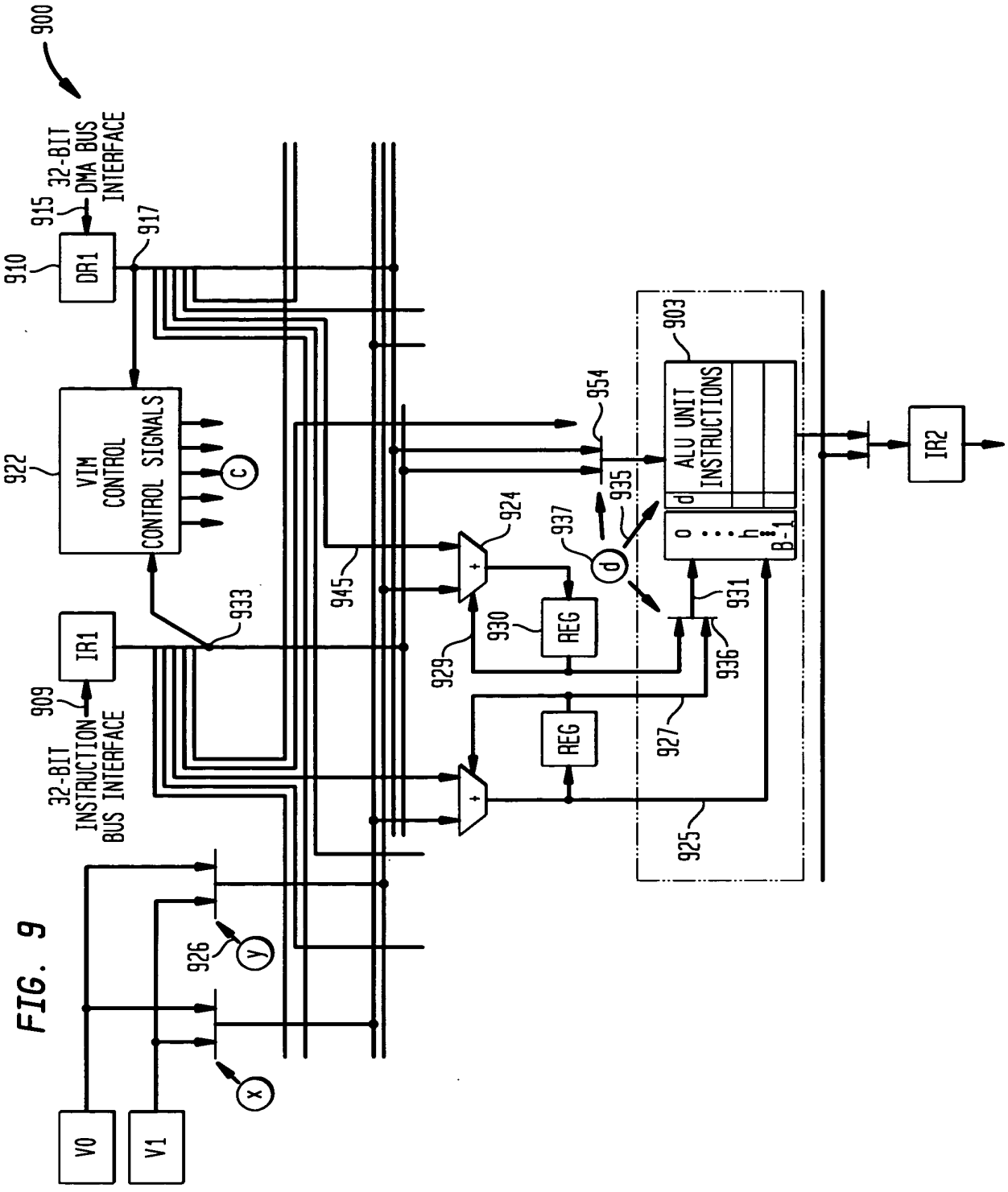

900
32-BIT DMA BUS INTERFACE
915
910
DR1
917
VIM CONTROL
CONTROL SIGNALS
922
C
954
903
ALU UNIT INSTRUCTIONS
d
0
h
B-1
IR2
924
937
935
945
933
930
REG
929
931
936
IR1
909
32-BIT INSTRUCTION BUS INTERFACE
927
REG
926
Y
X
925
V0
V1

FIG. 9